non-volatile media (e.g., optical or magnetic disks), volatile media (e.g., memory) and transmission media. Transmission media includes, without limitation, coaxial cables, copper wire and fiber optics.

[0068] Computer-readable mediums **812** or memory **813** can further include operating system **814** (e.g., Mac OS® server, Windows® NT server), network communication module **816**, real time beacon manager module **818** and beacon manager portal module **820**. Operating system **814** can be multi-user, multiprocessing, multitasking, multithreading, real time, etc. Operating system **814** performs basic tasks, including but not limited to: recognizing input from and providing output to devices **804**; keeping track and managing files and directories on storage devices **812***b* and memory **812***a;* controlling peripheral devices; and managing traffic on the one or more communication channels **810**. Network communications module **816** includes various components for establishing and maintaining network connections (e.g., software for implementing communication protocols, such as TCP/IP, HTTP, etc.). Beacon manager module **818** performs real time beacon commissioning and other beacon services as described in reference to FIGS. **1-6**. Beacon manage portal module **820** provides a web portal to enterprises as described in reference to FIGS. **1-6**. Beacon manage portal module **820** can also establish a web feed (e.g., XML feed) to beacon management service **112** from enterprise computer systems.

[0069] Architecture **800** can be included in any computer device, including one or more server computers each having one or more processing cores. Architecture **800** can be implemented in a parallel processing or peer-to-peer infrastructure or on a single device with one or more processors. Software can include multiple software components or can be a single body of code.

[0070] The features described may be implemented in digital electronic circuitry or in computer hardware, firmware, software, or in combinations of them. The features may be implemented in a computer program product tangibly embodied in an information carrier, e.g., in a machine-readable storage device, for execution by a programmable processor; and method steps may be performed by a programmable processor executing a program of instructions to perform functions of the described implementations by operating on input data and generating output.

[0071] The described features may be implemented advantageously in one or more computer programs that are executable on a programmable system including at least one programmable processor coupled to receive data and instructions from, and to transmit data and instructions to, a data storage system, at least one input device, and at least one output device. A computer program is a set of instructions that may be used, directly or indirectly, in a computer to perform a certain activity or bring about a certain result. A computer program may be written in any form of programming language (e.g., Objective-C, Java), including compiled or interpreted languages, and it may be deployed in any form, including as a stand-alone program or as a module, component, subroutine, or other unit suitable for use in a computing environment.

[0072] Suitable processors for the execution of a program of instructions include, by way of example, both general and special purpose microprocessors, and the sole processor or one of multiple processors or cores, of any kind of computer. Generally, a processor will receive instructions and data from a read-only memory or a random access memory or both. The essential elements of a computer are a processor for executing instructions and one or more memories for storing instructions and data. Generally, a computer may communicate with mass storage devices for storing data files. These mass storage devices may include magnetic disks, such as internal hard disks and removable disks; magneto-optical disks; and optical disks. Storage devices suitable for tangibly embodying computer program instructions and data include all forms of non-volatile memory, including by way of example semiconductor memory devices, such as EPROM, EEPROM, and flash memory devices; magnetic disks such as internal hard disks and removable disks; magneto-optical disks; and CD-ROM and DVD-ROM disks. The processor and the memory may be supplemented by, or incorporated in, ASICs (application-specific integrated circuits).

[0073] To provide for interaction with an author, the features may be implemented on a computer having a display device such as a CRT (cathode ray tube) or LCD (liquid crystal display) monitor for displaying information to the author and a keyboard and a pointing device such as a mouse or a trackball by which the author may provide input to the computer.

[0074] The features may be implemented in a computer system that includes a back-end component, such as a data server or that includes a middleware component, such as an application server or an Internet server, or that includes a front-end component, such as a client computer having a graphical user interface or an Internet browser, or any combination of them. The components of the system may be connected by any form or medium of digital data communication such as a communication network. Examples of communication networks include a LAN, a WAN and the computers and networks forming the Internet.

[0075] The computer system may include clients and servers. A client and server are generally remote from each other and typically interact through a network. The relationship of client and server arises by virtue of computer programs running on the respective computers and having a client-server relationship to each other.

[0076] One or more features or steps of the disclosed embodiments may be implemented using an Application Programming Interface (API). An API may define on or more parameters that are passed between a calling application and other software code (e.g., an operating system, library routine, function) that provides a service, that provides data, or that performs an operation or a computation.

[0077] The API may be implemented as one or more calls in program code that send or receive one or more parameters through a parameter list or other structure based on a call convention defined in an API specification document. A parameter may be a constant, a key, a data structure, an object, an object class, a variable, a data type, a pointer, an array, a list, or another call. API calls and parameters may be implemented in any programming language. The programming language may define the vocabulary and calling convention that a programmer will employ to access functions supporting the API.

[0078] In some implementations, an API call may report to an application the capabilities of a device running the application, such as input capability, output capability, processing capability, power capability, communications capability, etc.

[0079] As described above, some aspects of the subject matter of this specification include gathering and use of data available from various sources to improve services a mobile